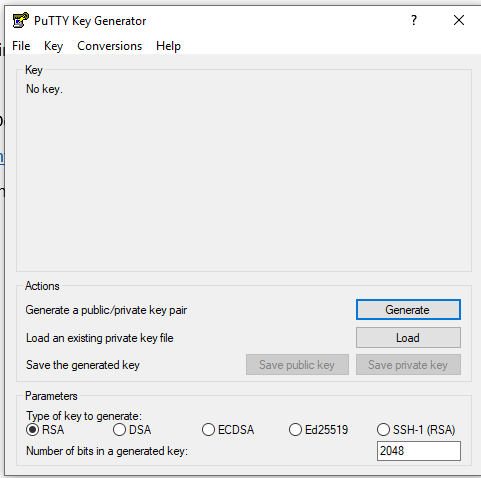
**1. Connecting to Remote Servers**

1.1. **Connecting from windows to a remote server(linux)**:

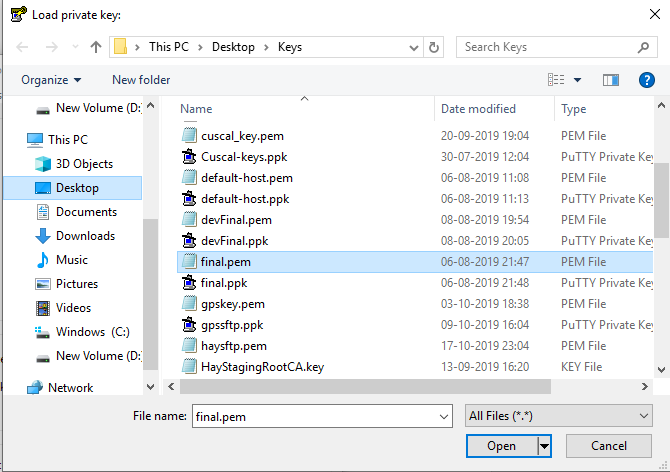
Step 1: Download Puttygen from the link below

<https://www.ssh.com/ssh/putty/windows/puttygen>

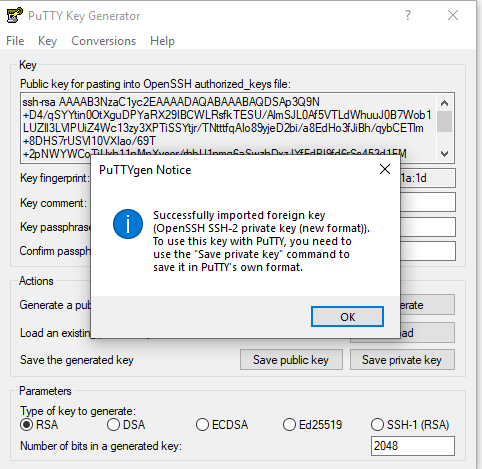
Step 2: Then open puttygen from the search bar



Step 3: then click on “Load” and select the .PEM file



Step 4: Then Click on “Open” next window will pop-up



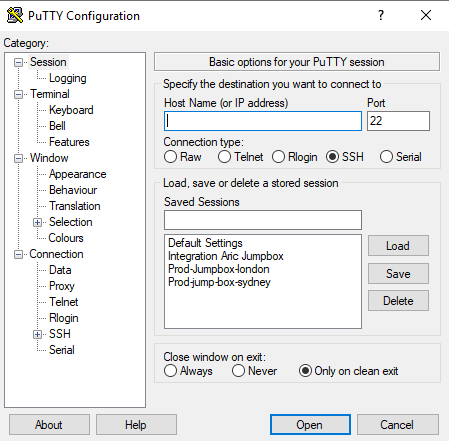
Then press on ok.

Step 5: Then Click on “Save Private Key” later we can use this private key to access the remote server using Putty.

Step 6. Download Putty from the link below and install

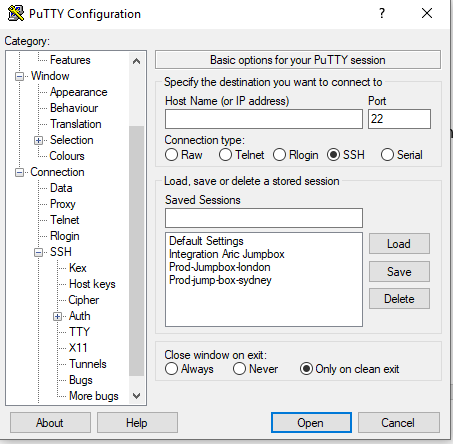
<https://www.putty.org/>

Step 7. Open Putty from the search bar

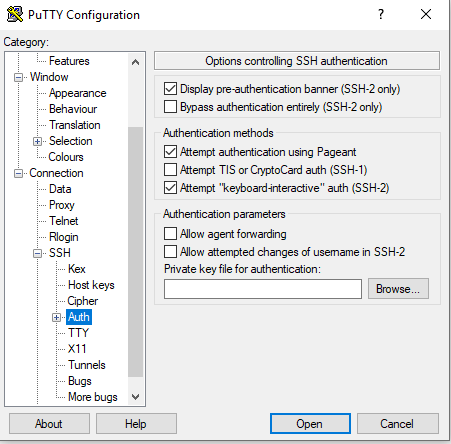


Step 8. In the Host name field specify host name or the IP address of the Server.

Step 9. Then Expand +SHH in the left hand bar

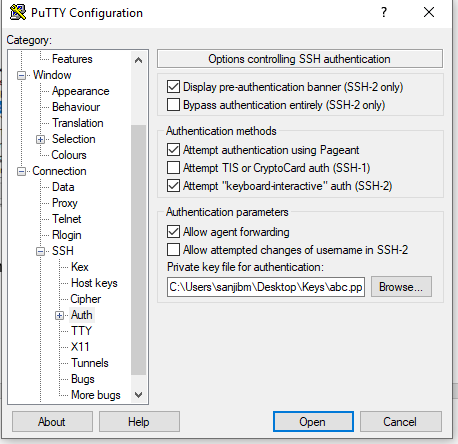


Step 10. Then click on “Auth”



Step 11. Then Click on “Browse”

Step 12: Then Select the specific “abc.ppk” file



Then you will go to login prompt. Specify ec2-user for RHEL server, specify centos/root for Centos server, Specify Ubuntu for linux Ubuntu flavour server as user and you will be able to login.

Note: If you have user ID and Password then directly you can specify the hostname or IP address in the hostname field and login using userID and Password.

2) **Connecting to Remote server from Mac/Ubuntu desktop**.

Step 1. At first get the PEM file like abc.pem and store it in a directory.

Step 2. Run Chmod command

Chmod 400 abc.pem

Step 3.Then run the below command

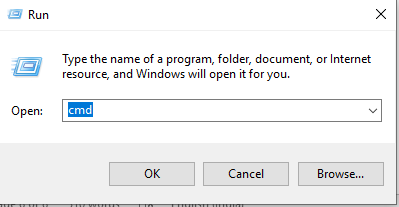
Ssh –i “abc.pem” ec2-user@server-ip or ssh –i “abc.pem” ec2-user@hostname

The user will be changed as the linux distro you are using like for RHEL it will be ec2-user, for centos it will be centos/root, for Ubuntu it will be Ubuntu etc.

**2. Configuring SSH keys**

Step 1. At first you have to generate a ssh key

Run ssh-keygen on the command prompt of your windows desktop.



Now you can choose different algorithm to generate your ssh keys like

ssh-keygen -t rsa -b 4096

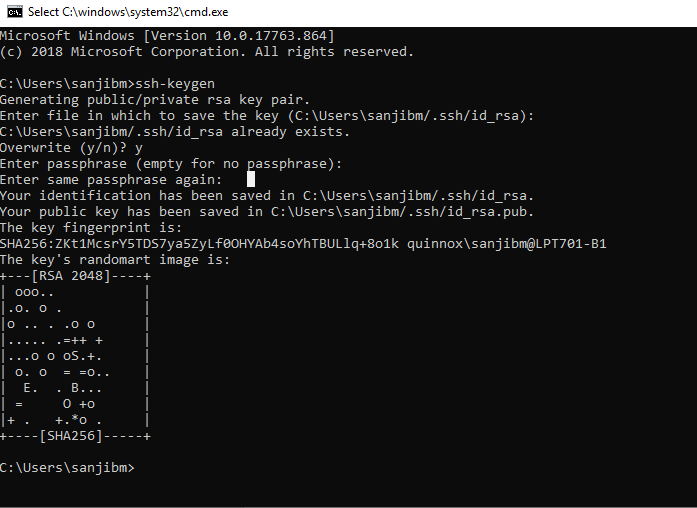
ssh-keygen -t dsa

ssh-keygen -t ecdsa -b 521

ssh-keygen -t ed25519

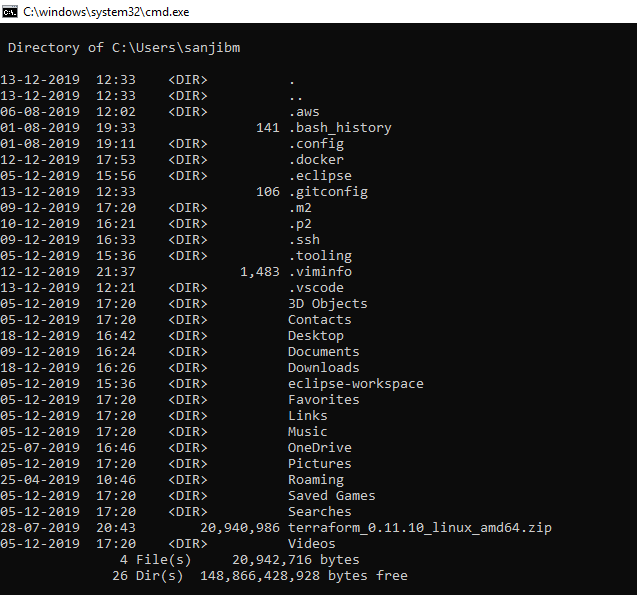
But mostly we use the default command”

Ssh-keygen (which uses rsa algorithm)

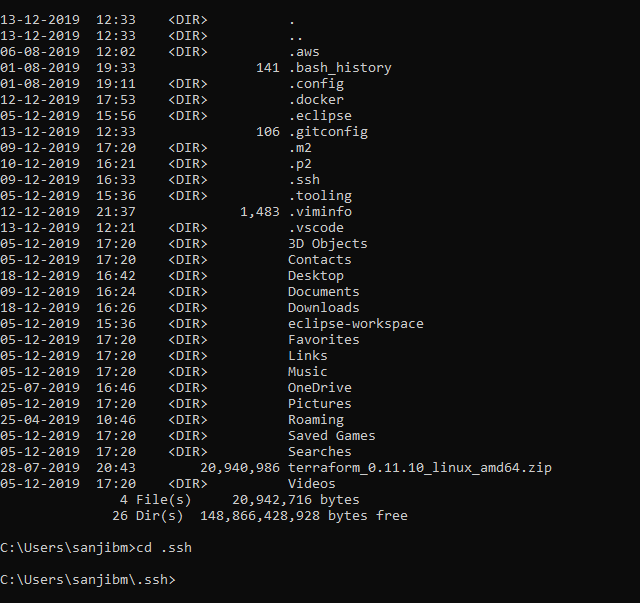


You can specify passphrase here to have some additional security.

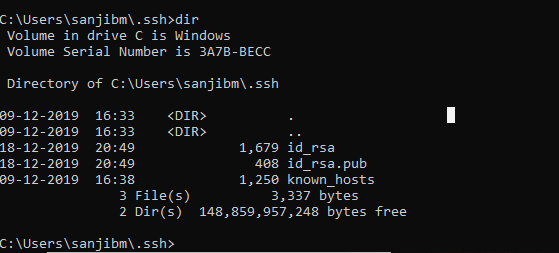
Step 2: then run “dir” on the command prompt



Step 3. Then go to .ssh folder by running “cd .ssh”



Step 4: run dir again



You will find your keys.

Now id-rsa.pub key has to be copied to server, inside authorized\_keys file.

And paste the content of id\_rsa.pub to authoried\_keys file of the default user and save&exit.

File location: /home/ec2-user/.ssh/authorized\_keys

You have to set proper permission for the directory and files,so that user can use the key.

Chmod 700 /home/<user-ID>/.ssh/

Chmod 600 /home/<user-ID>/.ssh/authorized\_keys

Then the user can login using his own private key.

Note:For Remote windows server you can use RDP to get connected

**3) How to use a custom key to configure openssh**

Step 1: Login to the remote server using default PUTTY or RDP using default user and “.PPK” file.

Note: for windows server you have to use RDP and administrative password by decoding it from the AWS console. User id will be” .\administrator”.

Step 2: Create user by running “useradd <ABC123>” command.

Step 3: Then switch to user ABC123

Su – ABC123

Step 4: you will get ABC123 prompt

ABC123@~

Step 5: Run ssh-keygen

Step 6: get the id\_rsa.pub key

Step 7: create .ssh directory

Mkdir .ssh

Chmod 700 .ssh

Step 8: CD to .ssh directory

Cd .ssh

Step 9: create file authorized\_keys

Touch authorized\_keys

Chmod 600 authorized\_keys

Step 10: open the file authorized\_keys with vi editor and paste the content of id\_rsa.pub file. Then save and exit.

Step 11: Get the private key in your working directory from you are login in.Then user can login

4) **Basics of ssh key generation and best practices**

For this go through

<https://www.ssh.com/ssh/ssh-key-basics>

5) **How to use ssh tunnelling to connect to other systems**

we define SSH port forwarding to forward a local port to the remote server.

In the below instructions, we are connecting to the sFTP server with the following details:

Host: 10.1.3.65

Port: 22

We map local port 10022 to the above remote server using the command when connecting using openssh under Windows 10 –L10022:10.1.3.65:22

When using PuTTY, you will need to configure SSH port forwarding under Connection/SSH/Tunnels:

And the click on Add, to get:

The current versions of Windows 10 have OpenSSH built into windows, and the same can be used to connect w/o using PuTTY. The instructions are given below for using OpenSSH in Windows 10 (the same commands will also work in a Mac or a Linux machine)

First run the below command:

ssh -i staging-key.pem ec2-user@3.104.109.106 -N -L10022:10.1.3.65:22 –v

Once the above command runs, then connect to the sftp server using the following details using any sftp client:

Host: localhost

Port: 10022

Username: Cuscalsftp

Password: Cuscalsftp

e.g.:

sftp –v -P 10022 Cuscalsftp@localhost

Do note that you can remove the –v from the ssh & sftp commands here to reduce the debug logging.